Procédure IMPR_RESU au format 'CASTEM'

1 But

Écrire un maillage et/ou les résultats d'un calcul sur un fichier au format 'CASTEM'.

Actuellement cette procédure permet d'écrire au choix :

- un maillage,
- des champs aux nœuds (de déplacements, de températures, de modes propres, de modes statiques, …),
- des champs par éléments aux nœuds (de contraintes, d'efforts généralisés, de variables internes…), si ces champs par éléments ne possèdent pas de sous points.

Pour les concepts de type resultat, on peut n'imprimer qu'une partie de l'information, en sélectionnant les champs et les numéros d'ordre que l'on désire exploiter.

1 Syntaxe

```
IMPR_RESU       (

◊  MODELE=  MO,                                         [modele]

   ♦    FORMAT   =    /    'CASTEM',

   ◊    UNITE    =    /    unit,                        [I]
                      /    37                           [DEFAUT]
   ◊    NIVE_GIBI    =    /    3,
                          /    10,                      [DEFAUT]

♦  RESU =(_F     (
   ♦    |         MAILLAGE      =    MA,        /       [maillage]
                                                /       [squelette]

        |    /    CHAM_GD       =    CH_GD,
             /    RESULTAT      =    RESU,

        # Extraction d'un champ de grandeur de resu
                  ◊    /        TOUT_CHAM    = /'OUI',    [DEFAUT]
                                               /'NON',
                       /        NOM_CHAM     = l_nomsymb, [l_K16]

                  ◊    /        TOUT_ORDRE   = 'OUI',     [DEFAUT]
                       /        NUME_ORDRE   = lordre,    [l_I]
                       /        LIST_ORDRE   = lenti,     [listis]
                       /        NUME_MODE    = lmode,     [l_I]
                       /        NOEUD_CMP    = lnoecmp,   [l_K16]
                       /        NOM_CAS      = ncas,      [l_K16]
                       /        ANGLE        = langl,     [l_K16]
                       /    /   FREQ         = lfreq,     [l_R]
                            /   LIST_FREQ    = lreel,     [listr8]
                            /   INST         = linst,     [l_R]
                            /   LIST_INST    = lreel,     [listr8]

                            ◊    | PRECISION =/ prec,       [R]
                                              / 1.0D-3,     [DEFAUT]
                                 | CRITERE   =/ 'RELATIF',  [DEFAUT]
                                              / 'ABSOLU',

        # Sélection des composantes
        ◊    /    TOUT_CMP          =    'OUI',             [DEFAUT]
             /    NOM_CMP           =    l_nomcmp,          [l_K8]

                   ),)  ,
)
```

2 Formats d'écriture et unité logique

La procédure IMPR_RESU permet d'écrire un maillage et/ou des valeurs de résultats au format du logiciel CASTEM 2000 développé par le CEA en vue d'une visualisation graphique par GIBI. Le logiciel GIBI (opérateurs de post-traitement de CASTEM 2000) est utilisable gratuitement avec *Code_Aster*.

On demande la création d'un fichier ASCII, qui pourra être relu par GIBI. Dans GIBI, la relecture du fichier se fait par les commandes OPTI REST FORM '***.cast' ; et REST FORM ; (les *** remplacent le nom du fichier de type cast donné par l'utilisateur dans astk).
UNITE vaut par défaut 37 et correspond au type cast dans astk.

```
◊       NIVE_GIBI      =    /    3,
                            /    10,          [DEFAUT]
```

L'opérande NIVE_GIBI permet de choisir le niveau GIBI d'impression du maillage et/ou des résultats. Le niveau 10 (par défaut) correspond à la version GIBI-98, mais les fichiers ainsi générés peuvent être relus par les versions plus récentes de GIBI. La version GIBI installée dans l'environnement Aster est la version GIBI-2000.

3 Mot clé RESU

Ce mot clé facteur permet de spécifier les résultats à imprimer et le format selon lequel on veut les imprimer.
Voir [U4.91.01].

4 Impression du maillage : opérandes MAILLAGE / MODELE

Cet argument permet d'imprimer le maillage au format 'CASTEM'. L'impression du maillage ne peut être demandée qu'une seule fois et avant tout autre résultat.

Le concept de type modele est facultatif dans la majorité des impressions.
Il est toutefois obligatoire, au format 'CASTEM', lors de l'impression d'un champ par élément (il est transféré en tant qu'objet CASTEM de type maillage).

Au format 'CASTEM', on imprime :

- pour chaque groupe de mailles *Aster* un objet CASTEM de type maillage et de même nom,
- un objet CASTEM de type maillage contenant toutes les mailles du maillage et accessible par le nom du maillage *Aster*,
- et, si l'opérande MODELE a été spécifié, un objet CASTEM de type maillage contenant toutes les mailles affectées dans le modèle *Aster* et accessible par le nom du modèle *Aster*.

Remarque :

Si le maillage a été généré avec Ideas ou GMSH, la numérotation Ideas ou GMSH n'est pas retranscrite au format CASTEM. La numérotation des nœuds dans CASTEM est identique à la numérotation des nœuds dans Aster.

1 Opérandes CHAM_GD et RESULTAT

Les valeurs des champs calculés sont stockées dans des structures de données appelées champs de grandeur. Ces champs de grandeur peuvent être directement accessibles (concept CHAM_GD) où se trouver dans une structure de données regroupant plusieurs champs de grandeur (concept resultat).

- un champ de grandeur est une structure de données qui permet de stocker des champs définis aux nœuds (cham_no_*) ou des champs définis par éléments (cham_elem_*). Pour les champs par éléments, on distingue les champs définis aux nœuds des éléments et les champs définis aux

points de GAUSS des éléments (ces derniers ne sont pas visualisables avec GIBI). Pour visualiser des champs par éléments aux points de GAUSS, l'utilisateur peut cependant utiliser MACR_ECLA_PG qui permet de construire un nouveau maillage en éclatant chaque maille de l'ancien maillage en n plus petites mailles et un champ aux nœuds avec des valeurs identiques en chaque nœud sommet de la maille.

- un concept resultat se compose d'un ou plusieurs champs de grandeur. Par exemple, à chaque pas de calcul on stocke dans la structure de données resultat, le champ de grandeur déplacement. Cette structure est indexée par une structure matricielle d'ordre 2, dont un indice est, par exemple, la liste des instants de calcul et l'autre l'ensemble des champs calculés (déplacements, contraintes, déformations, ...).

  On accède dans ce cas à un champ de grandeur en spécifiant une valeur d'une variable d'accès (numéro d'ordre, instant, fréquence, numéro de mode ...) et un nom de champ ('DEPL', 'SIGM_ELNO',...). Il existe plusieurs types de concept resultat : evol_elas, evol_noli, mode_meca,.... A chacun correspond une liste de champs et une liste de variables d'accès licites.

  Compte-tenu de la structure de données resultat, on comprend aisément que les possibilités d'impression dont on dispose sont celles des champs de grandeur, complétées par des possibilités spécifiques.

  Remarque :

  Au format 'CASTEM' on ne peut traiter que les champs aux nœuds ou les champs par élément définis aux nœuds (dans ce cas on doit obligatoirement spécifier le nom d'un modèle Aster par l'opérande MODELE).

Opérande CHAM_GD

Un champ aux nœuds *Aster* de nom u est retranscrit au format 'CASTEM' en un champoint de même nom. Dans CASTEM 2000, on peut donc accéder à ce champ par le nom u et on peut effectuer sur ce champ toutes les opérations permises sur des champoints. On notera cependant que les composantes des champs de grandeur DEPL qui sont dans *Aster* DX, DY, DZ, DRX, DRY, DRZ seront dans CASTEM UX, UY, UZ, RX, RY, RZ. Pour toutes les autres composantes le nom de la composante dans CASTEM sera défini par les quatre premiers caractères du nom de la composante dans *Aster*. Tous les champoints sont définis par rapport au maillage *Aster*.

Quelques opérations possibles dans GIBI sur un objet de type champoint et nommé u :

- Création et visualisation d'un objet de type déformée

```
DEF1 = DEFORME   MAILLAGE  U  AMPLI  COULEUR   ;
TRAC      DEF1   ;
```

- Création d'un objet de type vecteur

```
VEC1 = VECTEUR   U   UX UY UZ  AMPLI  COULEUR   ;
TRAC       VEC1      MAILLAGE  ;
```

- Création d'un objet de type champoint par extraction sur un champoint

```
DEPX = EXCO   UX  U   ;
TRAC           DEPX   MAILLAGE  ;
```

Un champ par élément aux nœuds de nom CHAM dans *Aster* est écrit au format 'CASTEM' en un chamelem de même nom. Dans CASTEM 2000, on peut donc accéder à ce champ par le nom cham et on peut effectuer sur ce champ toutes les opérations permises sur des chamelem aux nœuds. On notera cependant que les composantes des champs de grandeur SIGM qui sont dans *Aster* SIXX, SIYY, SIZZ, SIXY, SIYZ, SIXZ seront dans CASTEM SMXX, SMYY, SMZZ, SMXY, SMYZ, SMXZ. Les composantes de la grandeur VARI seront nommées V1, V2, ... Pour toutes les autres composantes le nom de la composante dans CASTEM sera obtenu par concaténation des deux premières et deux dernières lettres du nom de la composante dans *Aster*. Tous les chamelem aux nœuds sont définis par rapport au modèle *Aster*.

Quelques opérations possibles dans GIBI sur un objet de type cham_elem et nommé CHAM :

Tous les traitements sur des chamelem nécessitent la création préalable d'un objet CASTEM de type modele (rien à voir avec le modèle Aster) ce qui peut être fait par la commande :

```
MODCAST = MODE   MODELE MECANIQUE ELASTIQUE ;
```

- Création d'un objet de type chamelem par extraction sur un chamelem

```
SIGXX = EXCO  SMXX   CHAM   ;
TRAC   OEIL   SIGXX  MODCAST   MAILLAGE  ;
```

- Création d'un objet de type chamelem par opération sur un chamelem

```
TRE1   = TRESCA  CHAM   MODCAST   ;(contraintes de Tresca)
```

Opérande RESULTAT

L'opérande RESULTAT permet d'imprimer les champs contenus dans un concept resultat. On peut par exemple choisir de n'imprimer que certains champs (Cf. le mot clé suivant : NOM_CHAM).

Au format 'CASTEM', le concept resultat de nom TITI est écrit sous forme d'un objet CASTEM de type table (table) de même nom que le concept resultat *Aster*.

Le premier indice de la table est un entier i qui peut prendre la valeur des numéros d'ordre *Aster* (TITI . 1 est également une table).

La seconde table est indicée par une chaîne composée du nom symbolique *Aster* U = TITI . 1 . DEPL (champ de déplacements au numéro d'ordre 1), SIG = TITI . 2 . SIGM_ELNO (champ de contraintes au numéro d'ordre 2).

Exemple :

Dans *Aster* :

```
IMPR_RESU (RESU =(_F ( RESULTAT = TITI,
          NUME_ORDRE = (1,2),
          NOM_CHAM= ('DEPL','SIGM_ELNO')))
```

Dans CASTEM :

```
U   = TITI . 1 . DEPL  ; (objet CASTEM de type champoint)
SIG = TITI . 2 . SIGM_ELNO ; (objet CASTEM de type chamelem)
```

1 Extraction d'un champ de grandeur

Opérandes TOUT_CHAM / NOM_CHAM

Cf. document [U4.71.00].

Opérandes TOUT_ORDRE / NUME_ORDRE / LIST_ORDRE / NUME_MODE / INST / LIST_INST / FREQ / LIST_FREQ / NOEUD_CMP / NOM_CAS / ANGLE / PRECISION / CRITERE

Cf. document [U4.71.00].

2 Sélection sur les composantes

Une autre manière de réduire le volume des impressions est de n'imprimer que les valeurs de certaines composantes (par exemple que le déplacement suivant l'axe X : composante DX).

On ne peut pas imprimer de champs complexes au format CASTEM, il faut donc, pour de tels champs, sélectionner la PARTIE REELle ou IMAGinaire à imprimer.

Opérande TOUT_CMP

Ce mot clé permet d'indiquer que l'on désire imprimer toutes les composantes du champ. C'est la valeur par défaut.

Opérande NOM_CMP

Ce mot clé permet de choisir la liste des composantes du CHAM_GD (cham_no et/ou cham_elem) ou de tous les champs du concept résultat que l'on désire imprimer.

Les mots-clés TOUT_CMP et NOM_CMP ne peuvent être utilisés simultanément.

Ces composantes sont décrites dans la documentation spécifique des éléments.

3 Remarque

Dans Aster, il existe des champs par éléments avec des sous points (aussi bien pour des champs par éléments aux nœuds que pour des champs par éléments aux points de Gauss).

Au format 'CASTEM', IMPR_RESU ne permet pas l'impression de ces champs par éléments avec sous points.